

# **Building a Volume Logix and Open Symmetrix Manager (SymCLI) EMC Management Station**

By: Milton Vega  
(Rensselaer Technology Group, LTD.)

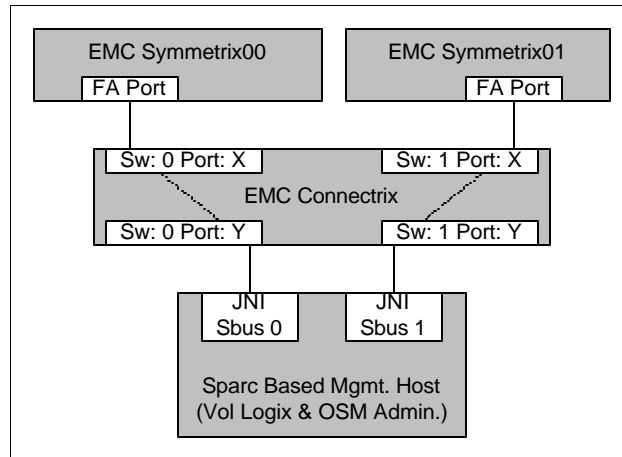
(07/2001)

## **Table Of Contents:**

<b>Preface</b>	Introduction & reference diagram
<b>Step.1</b>	Installing the JNI driver
<b>Step.2</b>	Configuring /kernel/drv/fcaw.conf
<b>Step.3</b>	Configuring /etc/system
<b>Step.4</b>	Rebuilding Solaris devices (to see JNI boards)
<b>Step.5</b>	Upgrading JNI fibre board Fcode
<b>Step.6</b>	Determining JNI fibre board WWNs
<b>Step.7</b>	Matching fcaw instances to physical board locations
<b>Step.8</b>	Creating ECFM zones for hba access to Symm. VCM database
<b>Step.9</b>	Configuring /kernel/drv/sd.conf
<b>Step.10</b>	Rebuilding Solaris devices (to see VCM hypervolumes)
<b>Step.11</b>	Installing the Volume Logix Software
<b>Step.12</b>	Initializing VCMDB's & granting access to GateKeepers
<b>Step.13</b>	Rebuilding Solaris devices (to see GateKeepers)

## Preface

This document details the procedures for customizing a SPARC/Solaris based machine so that it can be used to manage EMC Volume Logix Database(s), as well as run Open Symmetrix Manager (OSM) and the ECC SymmCLI command set. The diagram below shows how such a management station integrates into an EMC based Enterprise Storage Network. The box at the bottom of the figure represents the SPARC/Solaris based machine to which the customization instructions in this document apply. The document assumes that a patched O/S is already running, and that the custom EMC Fibre cards from Jaycor Networks Inc. (JNI) have already been installed. Further, it is also assumed, for purposes of illustration, that the system being customized is SBus based. The steps for PCI based systems are identical, but you'll have to replace SBus specific quantities, instructions, parameters etc., with its PCI counterpart.



**Step.1** The first step is to install the latest EMC supported version of the JNI driver for the FC64-1063 JNI fibre boards. Available as a SVR4 package from the JNI website, <http://www.jni.com>, it is installed using the pkgadd(1M) command as shown here:

```
root# pkgadd -d JNIfcawEMC.pkg
```

**Step.2** Next, we set specific parameters in the /kernel/drv/fcaw.conf file to values which depend on how other related components of this system has been configured.

```
# Set if EMC PowerPath is being used. This tells the fcaw driver to
# notify the sd driver of a failure within the timeout specified
# (in seconds), as opposed to waiting indefinitely or returning a busy
# status. This, in turn, will let the emcpower PowerPath driver know
# to initiate a failover.
failover = 60;

# Set the verbose flag so that fibre related diagnostic messages
# are echoed to the tty Console (in addition to being logged to the
# /var/admin/messages file). Default is 0.
fca_verbose = 1;

# For FC-SW switch fabric protocol.

# Set this to "1" if the HBA cards are connected to a EMC
# Connectrix fibre switch (instead of directly to a EMC
# Symmetrix FA port, or fibre hub). Set to 0 otherwise.
fca_nport = 1;

# Set this parameter to "fas" if you want to use persistent
# binding (the default is "fcaw*"). Persistent binding allows the
# user to determine the target each LUN will have, by specifying its
# value in the /kernel/drv/sd.conf file (Note: the Controller number is
# still determined by Solaris, and the LUN number still determined
# by the EMC bin file). Basically the fas token tells the kernel not
# to probe the SCSI chain in search of target information. Simply
# use the values specified in the /kernel/drv/sd.conf file.
def_hba_binding = "fas";

# For FC-AL Arbitrated Loop protocol; i.e. connections to a FA port,
# either directly or through a fibre hub. (Note: Persistent (target)
# binding is not possible with this scheme.) Is this true???

# For fibre connections directly to a FA port, or through a fibre hub.
fca_nport = 0;

# Any hba not explicitly bound in the sd.conf file
# will be allowed to probe all targets/LUN combinations.
def_hba_binding = "fcaw*";
```

**Step.3** Next, edit the /etc/system file and implement EMC and/or JNI recommended kernel parameter settings.

```
# The following kernel parameters are recommended in the EMC PowerPath
# manual and/or in the Jaycor driver instructions. Refer to the latest
# revisions of these documents for additions and/or modifications.

set sd:sd_io_time = 0x78
set sd:sd_max_throttle = 20
set scsi_options = 0x7f8
```

**Step.4** Next, perform a reconfiguration reboot to build Solaris devices for the JNI card instances.

```
root# reboot -- -r
```

**Step.5** The next step is to upgrade the Fcode firmware on the JNI fibre boards to the latest EMC/JNI supported revision, followed by another reboot. The latest supported Fcode revision, along with instructions on how to install it, are available on the JNI website, <http://www.jni.com>.

```
root# <implement Fcode upgrade instructions from JNI>.
...
root# reboot
```

**Step.6** At this point, we are ready to go to the Enterprise Connectrix Fabric Manager (ECFM) and perform the following tasks: (1) Verify that each hba card successfully logged into the Connectrix port to which it is attached, (2) correlate each hba instance (i.e. fcaw0, fcaw1,...) with the physical JNI fibre card and port it represents and, (3) create Connectrix zones which link hba ports to Symmetrix FA ports that have visibility to the VCM database and gatekeepers.

To do these things, we must first determine the **World Wide Number (WWN)** of each JNI card instance we have installed. We do so by searching for the string "fca" and "WWN" in the /var/adm/messages file, as shown here:

```
root# grep "fca" /var/adm/messages | grep WWN
Dec 3 11:45:57 emctest03 unix: fcaw0: Fibre Channel WWN: 200000e069c00618
Dec 3 11:45:57 emctest03 unix: fcaw0: Host: Port 201413 (WWN 200000e069c00618)
[...]
Dec 3 11:45:57 emctest03 unix: fcaw1: Fibre Channel WWN: 200000e06ac00210
Dec 3 11:45:57 emctest03 unix: fcaw1: Host: Port 201413 (WWN 200000e06ac00210)
```

From the output above, we see that instance 0 (**fcaw0**) of the fibre cards owns WWN "200000e069c00618", while instance 1 (**fcaw1**) of the fibre cards owns WWN "200000e06ac00210". Make a note of these associations, along with similar instance number to WWN number associations of any other JNI card instances that may exist. Armed with this information, we are prepared to go to the ECFM GIU and perform the steps noted above.

**Step.7** Either through fibre cable labels, or by cable tracing, establish where each hba port is connected to on the Connectrix. Once you have done that, it is easy to determine whether a hba port has successfully logged into its Connectrix port, by examining the status of that Connectrix port's LED on the ECFM GUI. A solid or blinking GREEN LED indicates successful login.

Clicking on that same port displays a properties dialogue box for it which, among other things, reveals that ports WWN, as well as the WWN of the hba port attached to it (also known as the attached port WWN). From this information, we can now associate the **fcaw instance number** to its **physical JNI card/port location**, by tracing the cable back to the hba card and port. (Note: if you familiar with Solaris device paths, and the POST probe order of the hardware, you can determine this relationship without the assistance of the ECFM GUI.) For your records, make a note of all fcaw instance number to JNI card/port location associations.

**Step.8** Next, for each hba port (JNI card), we create a Connectrix zone which links that hba port to the Symmetrix FA port which (via the EMC bin file) has been given visibility to the Volume Logix VCM database and gatekeepers. Generally each hba port will be used to exclusively communicate with, and manage its own Symmetrix, but this is not a requirement. A single hba can communicate with, and manage several Symmetrix devices as long as (through the Connectrix), that hba has visibility to the VCM database & GateKeeper hypervolumes of each of those Symms., and the sd.conf entries for that hba is properly configured (more on how to do that soon).

To properly set up the zones, you will need to gather the information specified in the following table for each Symmetrix. This information is usually provided in the EMC zoning port guide. (Note: The last four rows of information are actually not needed for the creation of zones; we request them here because they will be needed for tasks that come later on in the discussion).

Symm 00	Connectrix Sw # [0 1]	Connectrix Port #	Connectrix Port WWN#	Attached Port WWN#	FA Port #
VCM/GK FA Port					
HBA fcaw0					
	LUN #	Hyper ID#			
GateKeeper1 Hyper	240				
GateKeeper2 Hyper	241				
GateKeeper3 Hyper	242				
Vol Logix Hyper	0				

Symm 01	Connectrix Sw # [0 1]	Connectrix Port #	Connectrix Port WWN#	Attached Port WWN#	FA Port #
VCM/GK FA Port					
HBA fcaw1					
	LUN #	Hyper ID#			
GateKeeper1 Hyper	240				
GateKeeper2 Hyper	241				
GateKeeper3 Hyper	242				
Vol Logix Hyper	0				

Symm NN	Connectrix Sw # [0 1]	Connectrix Port #	Connectrix Port WWN#	Attached Port WWN#	FA Port #
VCM/GK FA Port					
HBA fcawN					
	LUN #	Hyper ID#			
GateKeeper1 Hyper	240				
GateKeeper2 Hyper	241				
GateKeeper3 Hyper	242				
Vol Logix Hyper	0				

Once the above information is compiled, you can create the necessary zones using the ECFM GUI. For Instructions on how this is done, refer to section 5.5 (page 122) of the **Enterprise Fibre Channel Director User's Guide** (P/N 300-600-004). Inclusion of those instructions here is beyond the scope of this document. (Perhaps an appendix however; TBD)

**Step.9** At this point, we need to make entries in the /kernel/drv/sd.conf file. These entries qualify what resource(s) a particular instance of a JNI device should probe for and, if present, how to configure a Solaris device node for it. (Note: it is legal, and almost always necessary, to have multiple sd.conf entries associated with the same JNI device.)

For example, the following entry tells instance 0 of the JNI device, **fcaw0**, to probe WWN address "**50060482bcc2edc2**" (a Symmetrix FA port in our case) for a disk device (hypervolume) assigned to LUN **4** (as set by the EMC bin file); if found, configure a Solaris device node for it with a target of **3**:

```
name="sd" class="scsi" target=3 lun=4 hba="fcaw0" wwn="50060482bcc2edc2";
```

So the Solaris device configured for this entry will look like "cWt3d4sZ". Note that, though other LUN's may exist at WWN address "**50060482bcc2edc2**" (say LUN 6), the configuration line above does not ask the fcaw0 instance to configure a device for it. That requires a separate configuration line. Similarly if, through the Connectrix switch fabric, fcaw0 had access to another LUN 4 device, but at a different WWN address (say "**50030482bc22edb7**"), no device would get configured for it either. Again, that would require a separate configuration line.

Note that the controller number (c#) of a device is determined by Solaris, while the LUN number (d#) of a device is determined by the EMC bin file. The target number(t#) is determined by the user via the "target=" sd.conf directive, and is a feature known as persistent binding.

Now that we understand the purpose of, and effect these sd.conf entries have on device configuration, we will go ahead and add entries that instruct specific JNI device instances to configure Volume Logix VCM and GateKeeper hypervolume devices for the Symmetrix being managed through that JNI device instance.

Below we provide an example of a system with two JNI fibre cards (two fcaw instances). Each fcaw instance is used to exclusively manage one of two Symmetrix. The FA port on Symm00 which has visibility to the Volume Logix database and Gatekeepers, is accessible by fcaw0. The WWN# of that FA port is "50060482bcc2edc2". Similarly, the FA port on Symm01 which has visibility to the Volume Logix database and Gatekeepers, is accessible by fcaw1. The WWN# of that FA port is "50030482bc22edb7". In both cases, the LUN number of the Volume Logix database is 0 (in fact, this is mandated by EMC engineering and is always set as such in the bin file). By convention, EMC begins the LUN numbers (d#) of the gatekeeper devices at 240, and increments from there. This strategy provides hinting common across all sites. Finally, we chose the user definable target for each of these LUN's to be 16. We choose 16 (or higher), to provide a hint that these are fibre devices, since traditional non-fibre SCSI devices can only assume target values from 0 to 15, inclusive.

```
# JNI Fibre Controller 0
# Entry so that the host sees the Volume Logix database. The Volume Logix
# database hypervolume LUN is set by EMC, and is always 0.
# Persistent binding allows the Target to be user defined,
# but it is usually set to 16 or above for hinting purposes.
name="sd" class="scsi" target=16 lun=0 hba="fcaw0" wwn="50060482bcc2edc2";

# Entry for gatekeepers which OpenSymm Manager will use.
name="sd" class="scsi" target=16 lun=240 hba="fcaw0" wwn="50060482bcc2edc2";
name="sd" class="scsi" target=16 lun=241 hba="fcaw0" wwn="50060482bcc2edc2";

# JNI Fibre Controller 1
# Entry so that the host sees the Volume Logix database. The Volume Logix
# database hypervolume LUN is set by EMC, and is always 0.
# Persistent binding allows the Target to be user defined,
# but it is usually set to 16 or above for hinting purposes.
name="sd" class="scsi" target=16 lun=0 hba="fcaw1" wwn="50030482bc22edb7";

# Entry for gatekeepers which OpenSymm Manager will use.
name="sd" class="scsi" target=16 lun=240 hba="fcaw1" wwn="50030482bc22edb7";
name="sd" class="scsi" target=16 lun=241 hba="fcaw1" wwn="50030482bc22edb7";
```

Once you have completed editing the sd.conf file, you can perform a syntax check using the "canal" utility that comes with the JNI fcaw package (JNIfcawEMC.pkg) located in /opt/JNIfcaw. It's usage is as follows:

```
root# /opt/JNIfcaw/canal /kernel/drv/sd.conf
```

**Step.10** Perform a reconfiguration reboot to configure Solaris device nodes for the VCM hypervolumes.

```
root# reboot -- -r
```

(**Note:** The gatekeepers are not visible at this stage; however, we have made the necessary sd.conf modification for them in advance, so we are prepared when they are finally made visible by Volume Logix).

Once the reboot completes, check the /var/adm/messages file to verify that each hba instance successfully logged into the fabric and that the link is operational; then check the output of the format(1M) command (or the /dev/(r)dsk directory) to see if device nodes for the VCM hypervolumes were created (again, the gatekeeper devices are not available at the stage):

```
root# egrep "fcaw0|fcaw1" /var/adm/messages
[...]
```

```
Dec  3 11:45:57 emctest03 unix: fcaw0: Target 16 Lun 0: Port 201e13
(WWN 50060482bcc2edc2) ready.
[...]
```

```
Dec  3 11:45:57 emctest03 unix: fcaw1: Target 16 Lun 0: Port 206e14
(WWN 50030482bc22edb7) ready.
[...]
```

```
root# format
[...]
```

```
4. c2t16d0 <EMC-SYMMETRIX-5265 cyl 14 alt 2 hd 15 sec 64>
   /sbus@4,4000/fibre-channel@2/sd@11,0
[...]
```

```
6. c3t16d0 <EMC-SYMMETRIX-5265 cyl 14 alt 2 hd 15 sec 64>
   /sbus@4,4000/fibre-channel@2/sd@11,0
[...]
```

```
<Label the Volume Logix Database Hypervolumes Now!!!>
```

As with any other EMC hypervolume, the Volume Logix database hypervolumes will not be labeled when they first appear. The Volume Logix software (i.e. fpath) does not care whether the Volume Logix database hypervolumes are labeled, but we label them anyway to eliminate the "Wrong Magic Number" system warning messages associated with non labeled disks. **Be sure to label them now.** Once you initialize the VCM database with the "fpath initdb" command (which we'll get to in a moment), the VCM database hypervolumes become read-only to everything except the Volume Logix "fpath" command (which means you'll no longer be able to label those disks; Volume Logix will work fine, but you'll have to live with those pesky "Wrong Magic Number" messages forever).

**Step.11** We are ready to install the Volume Logix Database Management Software. Currently it ships as the **tar** file **admin.solaris.2.0.tar**. Assuming the tar file exists in /tmp, you can install it using the following procedure:

```
root# mkdir -p /opt/symmapps/vcm
root# cd /opt/symmapps/vcm
root# tar -xvfp /tmp/admin.solaris.2.0.tar
```

**Step.12** Finally we can initialize the Volume Logix database and begin to use Volume Logix to authenticate communications between host hba's (i.e. specific JNI ports on specific hosts) and hypervolumes (i.e. specific hypervolume, on specific FA ports, on specific Symms.).

The first step is to initialize the Volume Logix database on each Symm:

```
root# cd /opt/symmapps/vcm
root# fpath initdb -d c2t16d0s2
[...]
root# fpath initdb -d c3t16d0s2
[...]
```

**Note:** This is a very destructive command. It deletes ALL Symm. hypervolume to hba authentication associations previously created. Use with caution.

Next, we issue fpath commands to authenticate communications between certain host hba's and certain Symmetrix hypervolumes over certain Symmetrix FA ports. Our first stop?... (1) authenticating communication between fcaw0 on this machine and GateKeeper hypervolumes on the Symm00 FA port set aside for that fcaw0 port and (2) authenticating communication between fcaw1 on this machine and GateKeeper hypervolumes on the Symm01 FA port set aside for that fcaw1 port. Assuming that this Symm00 FA port number is **14b**, and that this Symm01 FA port number is **13b**, and that the respective GateKeeper hypervolume ID's are identified in the following table,

Symm 00	LUN #	Hyper ID#
GateKeeper1 Hyper	240	155
GateKeeper2 Hyper	241	156
GateKeeper3 Hyper	242	157

Symm 01	LUN #	Hyper ID#
GateKeeper1 Hyper	240	178
GateKeeper2 Hyper	241	179
GateKeeper3 Hyper	242	180

here is how it is done:

```
root# ./fpath adddev -d /dev/rdsk/c2t16d0s2 -w 200000e069c00618 \
-f 14b -r "155-157"
root# ./fpath adddev -d /dev/rdsk/c3t16d0s2 -w 200000e06ac00210 \
-f 13b -r "178-180"
root# ./fpath refresh -d /dev/rdsk/c2t17d0s2; (commit changes)
root# ./fpath lsdb -d /dev/rdsk/c2t17d0s2 (confirm changes)
```

**Step.13** Perform a reconfiguration reboot to configure devices for the (now accessible) GateKeeper devices.

```
root# reboot -- -r
```

With GateKeepers now available to the system, we can now install the EMC Open Symmetrix Manager base and add-on components, as well as any other EMC management software that use GateKeepers.